Notice of Allowability	**Application No.** 09/611,403	**Applicant(s)** RINGSETH ET AL.	
	Examiner William H. Wood	**Art Unit** 2193	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *30 March 2006*.

2. ☒ The allowed claim(s) is/are *1, 4-9, 16-17, 22-26 and 37-42*.

3. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☐ All b) ☐ Some* c) ☐ None of the:
   1. ☐ Certified copies of the priority documents have been received.
   2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
   3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
   1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
   (b) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☐ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☐ Information Disclosure Statements (PTO-1449 or PTO/SB/08), Paper No./Mail Date _____
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application (PTO-152)
6. ☐ Interview Summary (PTO-413), Paper No./Mail Date _____ .
7. ☒ Examiner's Amendment/Comment
8. ☒ Examiner's Statement of Reasons for Allowance
9. ☐ Other _____.

EXAMINER'S AMENDMENT

An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Kyle B. Rinehart on 12 June 2006.

The application has been amended as follows:

Claim 1

A computer readable medium having stored thereon a computer executable compiler system that performs semantic analysis of interface definition language constructs ~~information~~ embedded in programming language code in a file, the compiler system comprising:

a front end module that separates a file into plural tokens, the file including programming language code having embedded therein interface definition language constructs ~~information~~;

a converter module that converts the plural tokens into an intermediate representation, wherein the intermediate representation includes a symbol table and a tree that unifies representation of the programming language code and the embedded interface definition language constructs ~~information~~, wherein at least some of the embedded interface definition language constructs are ~~information is~~ represented in the tree without creating new programming language code for the at least some of the embedded interface definition language constructs ~~information~~, wherein the symbol table includes plural entries for symbol names for the programming language code, and wherein at least one of the plural entries has an associated list of definition language attributes; and

a back end module that produces output computer-executable code from the intermediate representation based at least in part upon semantics of the embedded interface definition language constructs ~~information~~.

Claim 4

The compiler system of claim 1 further comprising a definition language attribute provider that modifies the intermediate representation based upon the semantics of the embedded interface definition language constructs ~~information~~.

Claim 6

~~In a computer system, a~~ A computer executable compiler system stored in a computer system that creates a unified programming language and interface definition language parse tree from a file comprising a mix of programming language constructs and interface definition language constructs, the compiler system comprising:

a front end module that separates a file into plural tokens, the file comprising a mix of programming language constructs and interface definition language constructs; and

a converter module that converts the plural tokens into an intermediate representation comprising a symbol table and a parse tree, wherein:

the symbol table includes plural entries for symbol names for the programming language constructs, at least one of the plural entries having an associated list of interface definition language attributes;

the parse tree unifies representation of the programming language constructs and the interface definition language constructs; and

at least some of the interface definition language constructs are represented in the parse tree without creating new programming language constructs for the at least some of the interface definition language constructs.

Claim 9

A computer readable medium having stored thereon a data structure representing a unified interface definition language and programming language parse tree for a file having a combination of programming language code and embedded interface definition language constructs ~~information~~, the data structure comprising:

a first data field storing data representing a symbol table that has plural entries, each of the plural entries corresponding to a symbol name for programming language code of a file having a combination of programming language code and embedded interface definition language constructs ~~information~~, at least one of the plural entries having an associated list of interface definition language attributes based upon the embedded interface definition language constructs ~~information~~; and

a second data field storing data representing a parse tree, wherein the parse tree unifies representation of the programming language code and the embedded interface definition language constructs ~~information~~; and

wherein at least some of the embedded interface definition language constructs are ~~information is~~ represented in the parse tree without creating new programming language code for the at least some of the embedded interface definition language constructs ~~information~~.

Cancel claims 10-12; 13-15.

Claim 16

A computer readable medium having stored thereon instructions for performing a method of creating a unified programming language and definition language tree from a file that includes interface definition language constructs ~~information~~ embedded in programming language code, the method comprising:

separating a file into plural tokens, the file including interface definition language constructs ~~information~~ embedded in programming language code;

building a symbol table having plural entries for symbol names for the programming language code, at least one of the plural entries having an associated list of definition language attributes based upon the embedded interface definition language constructs ~~information~~; and

building a tree that unifies representation of the embedded interface definition language constructs ~~information~~ and the programming language code, wherein the building comprises representing at least some of the embedded interface definition language constructs ~~information~~ in the tree without creating new programming language code for the at least some of the embedded interface definition language constructs ~~information~~.

Claim 17

The computer readable medium of claim 16 wherein the separating comprises recognizing a delimiting character that distinguishes definition language tokens from programming language tokens.

Cancel claims 18-19; 20; 21.

Claim 22

The compiler system of claim 1 wherein the backend module also produces output interface definition language information in an output file that includes the output computer-executable code.

Claim 23

The compiler system of claim 1 wherein the backend module also produces output interface definition language information in a separate output file from the output computer-executable code.

Claim 26

The compiler system of claim 1 wherein the programming language code is in C++ and wherein the embedded interface definition language constructs ~~information~~ includes IDL constructs.

Cancel claims 27-31; 32-36

Claim 37

The method of claim 1 wherein the embedded interface definition language constructs ~~information~~ includes an export attribute, wherein the export attribute annotates a user-defined data type, and wherein the compiler system outputs interface definition language metadata for the user-defined data type based at least in part upon the export attribute.

Claim 38

The method of claim 1 wherein the embedded interface definition language constructs ~~information~~ includes an interface type attribute, wherein the interface type attribute annotates an interface, and wherein the output computer-executable code includes implementation code for an implementation of the interface.

Claim 41

The method of claim 1 wherein the embedded interface definition language constructs ~~information~~ includes a project attribute.

Claim 42

The method of claim 1 wherein a definition language attribute provider reacts to plural events during compilation of the file by causing modification of the intermediate representation, wherein at least one of the plural events, in reaction to which the definition language attribute provider causes modification of the intermediate representation, occurs during processing of the embedded interface definition language constructs ~~information~~, and wherein at least one of the plural events, in reaction to which the definition language attribute provider causes modification of the intermediate representation, occurs during processing of the programming language code.

Reasons for Allowance

Claims 1, 4-9, 16-17, 22-26 and 37-42 are allowed.

The following is an examiner's statement of reasons for allowance: the prior art of record fails to teach or suggest the claimed invention. Specifically, the prior art of record fails to teach or suggest embedded interface definition language constructs and a compiler system such that the symbol table includes plural entries for symbol names for the programming language code and wherein at least one of the plural entries has an associated list of definition language attributes, as recited in the independent claims.

Having considered Applicant's Exhibits A-C provided in response to the previous Requirement for Information, the **Grimes** reference is withdrawn from rejection. The prior art of record, **Grimes**, "Attribute Programming", disclosed attributes being embedded in code. However, **Grimes** failed to disclose the independent claims as recited, including a unified representation with the described symbol table.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Correspondence Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to William H. Wood whose telephone number is (571)-272-3736. The examiner can normally be reached 9:00am - 5:30pm Monday thru Friday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki can be reached on (571)-272-3719. The fax phone numbers for the organization where this application or proceeding is assigned are (571)273-8300 for regular communications.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained form either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR systems, see http://pair-direct.uspto.gov. For questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703)305-3900.

William H. Wood
Patent Examiner
AU 2193
June 12, 2006

KAKALI CHAKI
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100